

FIG. 1 Normal Subject Controls and Patient Selection for Development of the Cognitive Decline Index

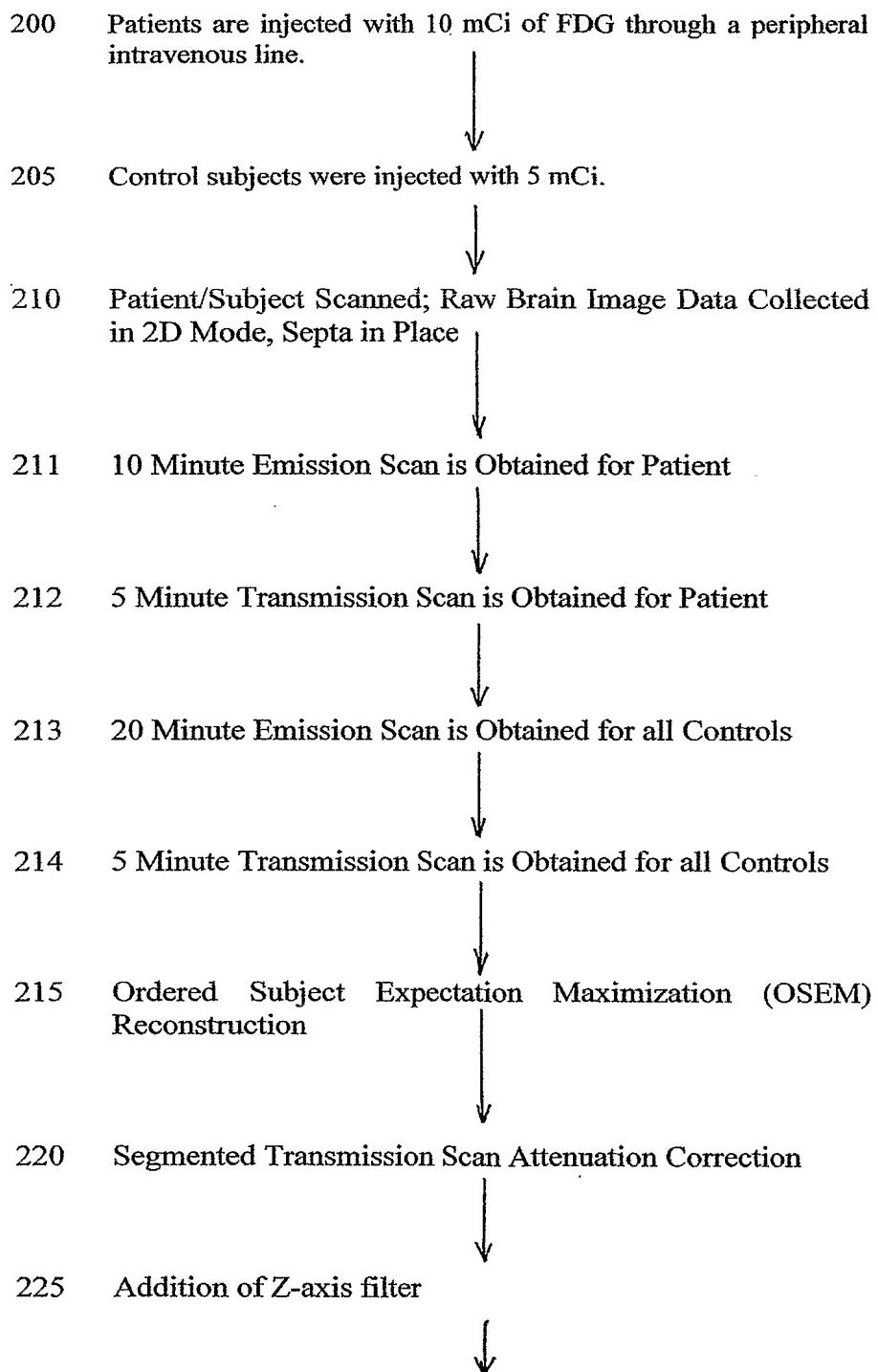


FIG. 2a Derivation of the Processed Digital Brain Image

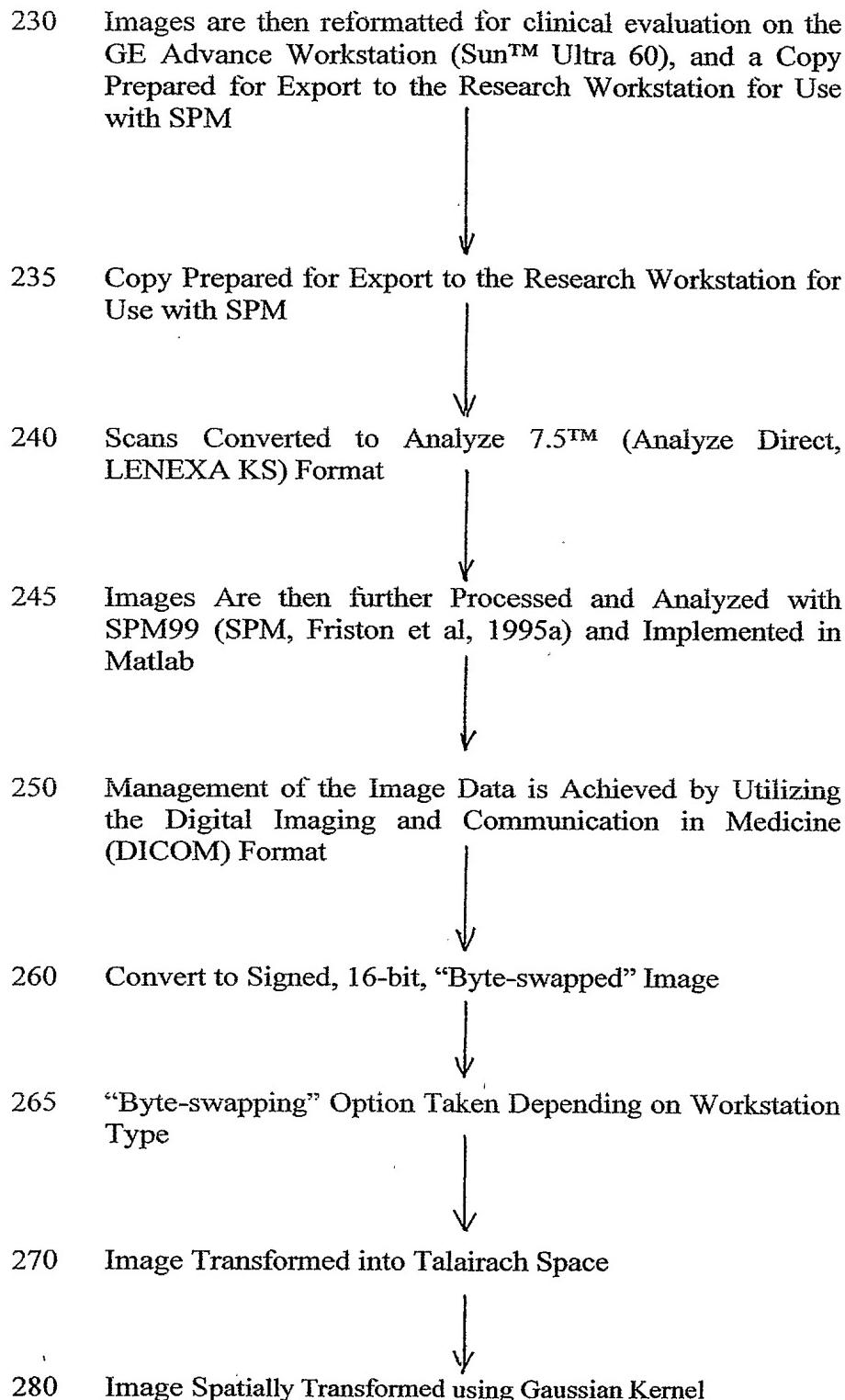
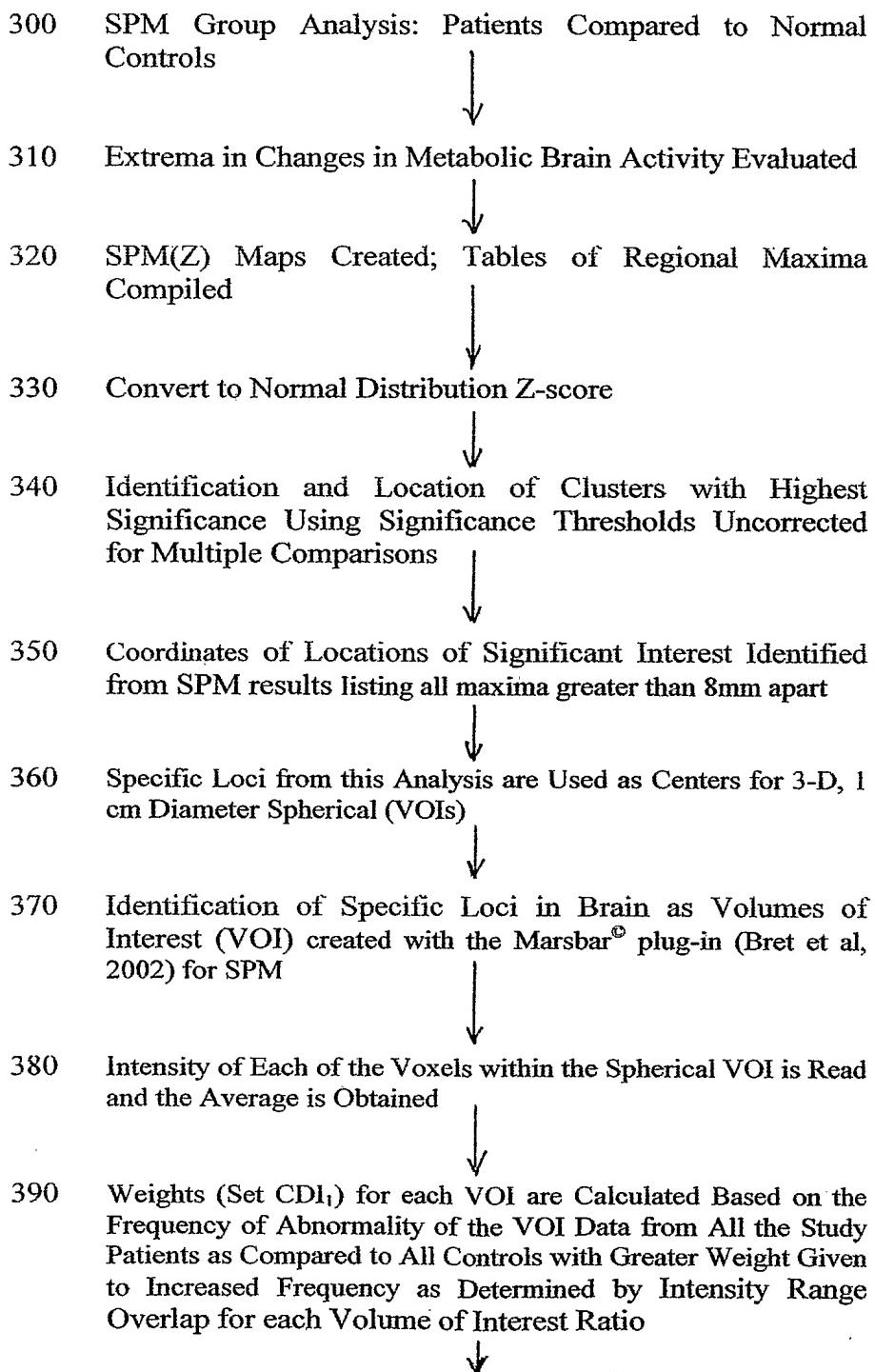


FIG. 2a (Cont'd) Derivation of the Processed Digital Brain Image

**FIG. 2b Derivation of Region Location and Identification of VOIs**

- 391 Import VOI Data Into Spreadsheet
- 392 Determine Intensity Range Overlap for each VOI Ratio
- 393 Create Weights for each Intensity Extreme
- 394 Create Weighted VOI Ratio
- 395 Scale and Normalize Ratio
- 400 Calculation of CDI:

$$CDI = C_x + \left[\sum_{j=1}^n V_j X_j / n \right] / \left[\sum_{i=1}^m W_i Y_i / m \right]$$

Where X_j denotes the j^{th} Increased Intensity Value;
 V_j denotes the j^{th} Weight for the j^{th} Increased Intensity Value;

Y_i denotes the i^{th} Decreased Intensity Value; and
 W_i denotes the i^{th} Weight for the i^{th} Decreased Intensity Value.

C_x is the correction factor used to normalize the dataset.

- 410 Weights of Set CDI_1 are then used as a baseline for calculation of a second set of Weights (Set CDI_2) to calculate CDI_2 . Set CDI_2 is calculated by iterative optimization of each weight to maximally separate the patient from the controls

FIG. 2b (Cont'd) Derivation of Region Location and Identification of VOIs

- 500 CDI is Compared to Established Normal Range of Values; Presence of Normality or Abnormality is Determined
- 510 If CDI Reading is Negative, Patient Educated About the Clinical Course of Potential Illnesses and Signs to Watch out for; Potential Benefit of Preventative Measures such as Antioxidants, Brain Exercises, Beneficial Diet and Adequate Rest are Discussed
- 520 If CDI Reading is Positive, Patient Educated about the Meaning of a Positive CDI Reading, Projected Clinical Course of Illness, Benefit of Medication, Potential Benefit of Preventative Measures such as Antioxidants, Brain Exercises, Beneficial Diet and Adequate Rest are Discussed
- 530 Following either Step 620 or Step 630, Results are Given to Referring Physician and Patient Scheduled for Re-evaluation in one year.
- 540 Patient Data is Stored in the Comprehensive Patient Database

FIG. 2c Patient Diagnosis and Clinical Recommendations

Fig. 2D Construction of the Neural Network Based CDI (CDInn)

- 600 Append the Neural Network Toolbox to the Matlab Path
 ↓
- 610 Type the command “nnntool” at the Matlab command prompt to start the NN graphical user interface
 ↓
- 620 Configure an artificial neural network with the features and parameters:
 Name: CDInn
 Type: Feed-forward back-propagation
 Input ranges: [-1 1; -1 1; -1 1; -1 1; -1 1; -1 1; -1 1; -1 1; -1 1;
 Training function: trainlm
 Adaption learning function: Learndgm
 Performance function: MSE
 Layers: 2
 Number of neurons: 2
 Transfer function neuron 1: purelin
 Transfer function neuron 2: tansig
 ↓
- 630 VOI regions obtained from the specified locations:
 Regions of increased activity:
vermis,
motor,
R pons, and
cerebellar nuclei
- Regions of decreased activity:
posterior cingulate,
L parietal,
R parietal,
L temporal01, and
L temporal02
 ↓
- 640 Intensity data from the above regions are entered into the NN in the order:
vermis,
motor,
R pons,
cerebellar nuclei,
posterior cingulate,
L parietal,
L temporal01,
L temporal02,
R parietal.
 ↓
- 650 VOI Data are mean-normalized by subject, i.e. divided by the average of the 9 regions measured for each subject. This gave each subjects data set a mean value =1.
 ↓
- 660 The datasets are next divided into comparably sized subgroups (c1 denotes control subgroup # 1, p1 = patient subgroup #1, etc): [c1=16, c2=17, p1=16, p2=16].
 ↓

670 Four mixed groups were made from these by pairing controls with patients to give: [c1p1,
c1p2, c2p1, c2p2].
 ↓

680 These datasets are then exported (from Excel) to a text file, subsequently imported into
Matlab and converted to a .mat file for use by Matlab.
 ↓

690 Rescaling is performed to ensure the input range is appropriately matched for the tansig
function on the output layer of the NN, which will classify the input data in the range ± 1 .
This is carried out in the main Matlab window at the command prompt. As an example,
the dataset c1p1 is rescaled using the premnmx function:

[c1p1_rescale,minp,maxp]=premnmx(c1p1)

↓

700 The rescaled output from this function is labeled and saved as **c1p1_rescale.mat**. This is
done for each of the four paired datasets listed above.
 ↓

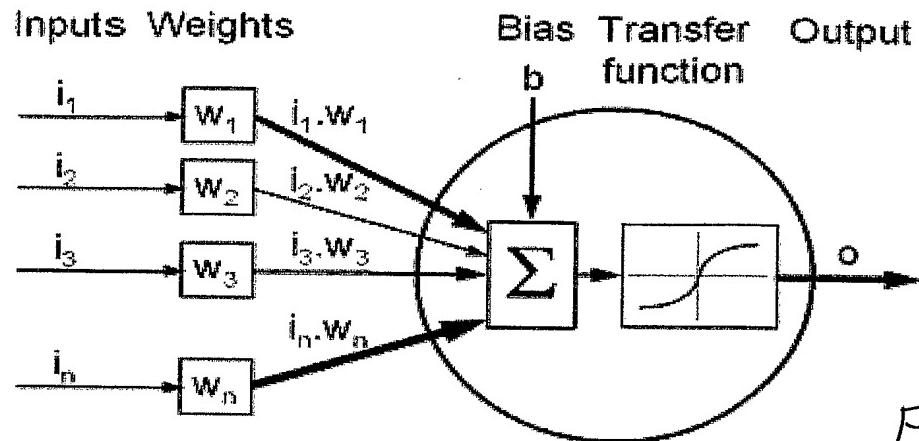
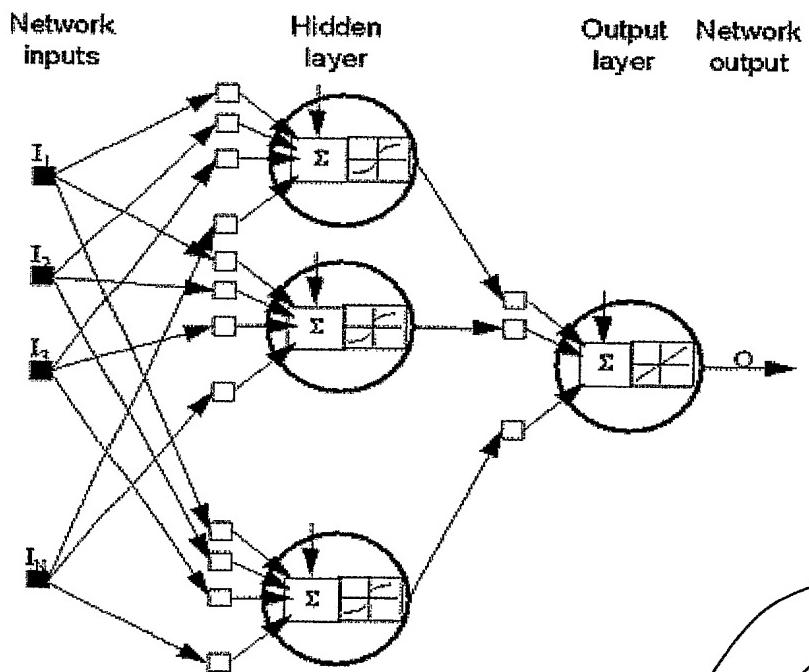
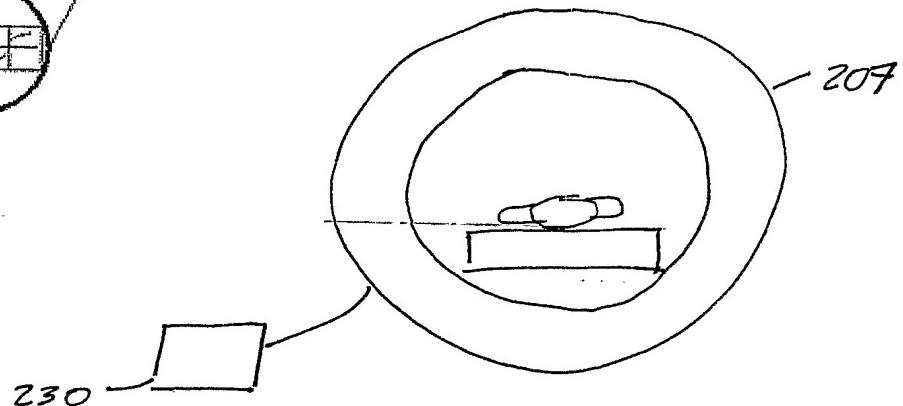
710 These four files are then used as “inputs” for neural net training. For training of the
neural net, each input file had a “target” file to which the neural network tried to match
the input dataset to. These files were labeled to match the dataset from which they were
used, thus for **c1p1_rescale.mat**, the “target” file was labeled **c1p1t.mat**. Each target file
consisted of a single row of digits with the same number of columns as the input file. For
each patient (column) in the **c1p1_rescale** dataset, a “-1” was present in the **c1p1t.mat**
file, and for each control, a “1” was assigned. Thus, the four input and target datasets
were:
 ↓

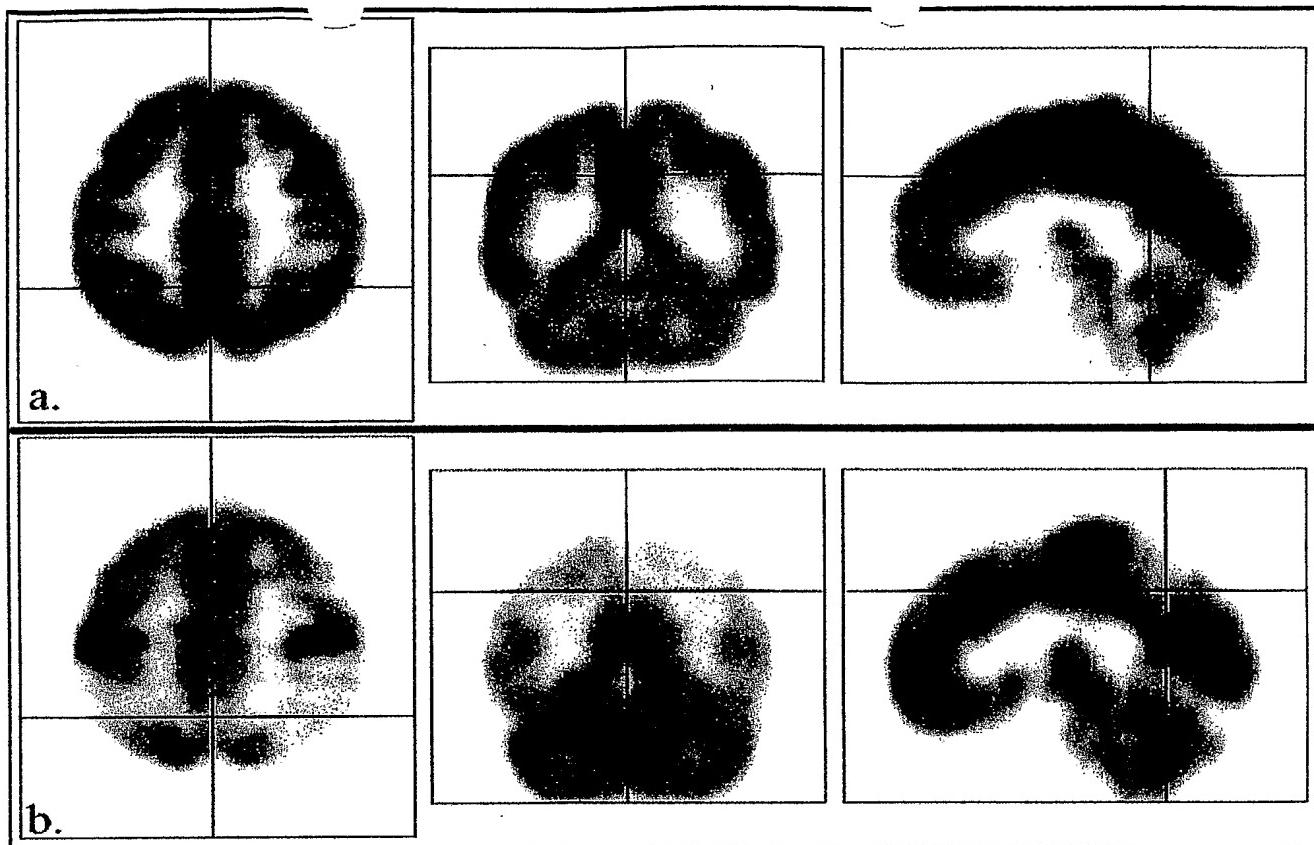
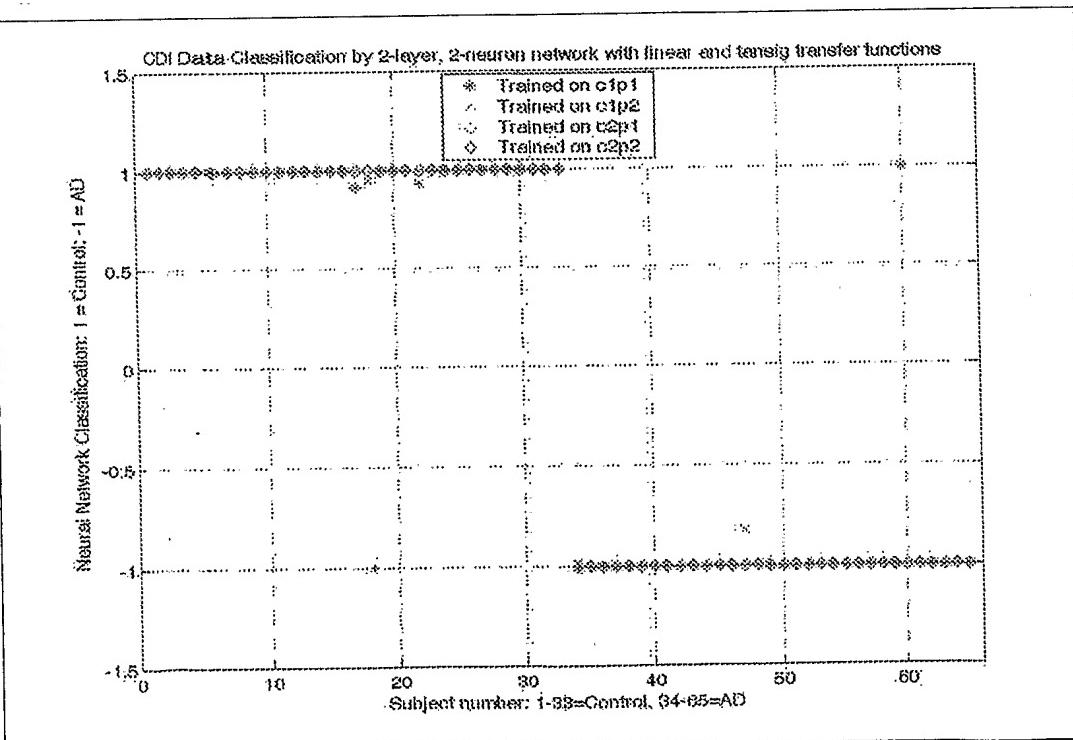
Input	Target
c1p1_rescale.mat	c1p1t.mat
c1p2_rescale.mat	c1p2t.mat
c2p1_rescale.mat	c2p1t.mat
c2p2_rescale.mat	c2p2t.mat

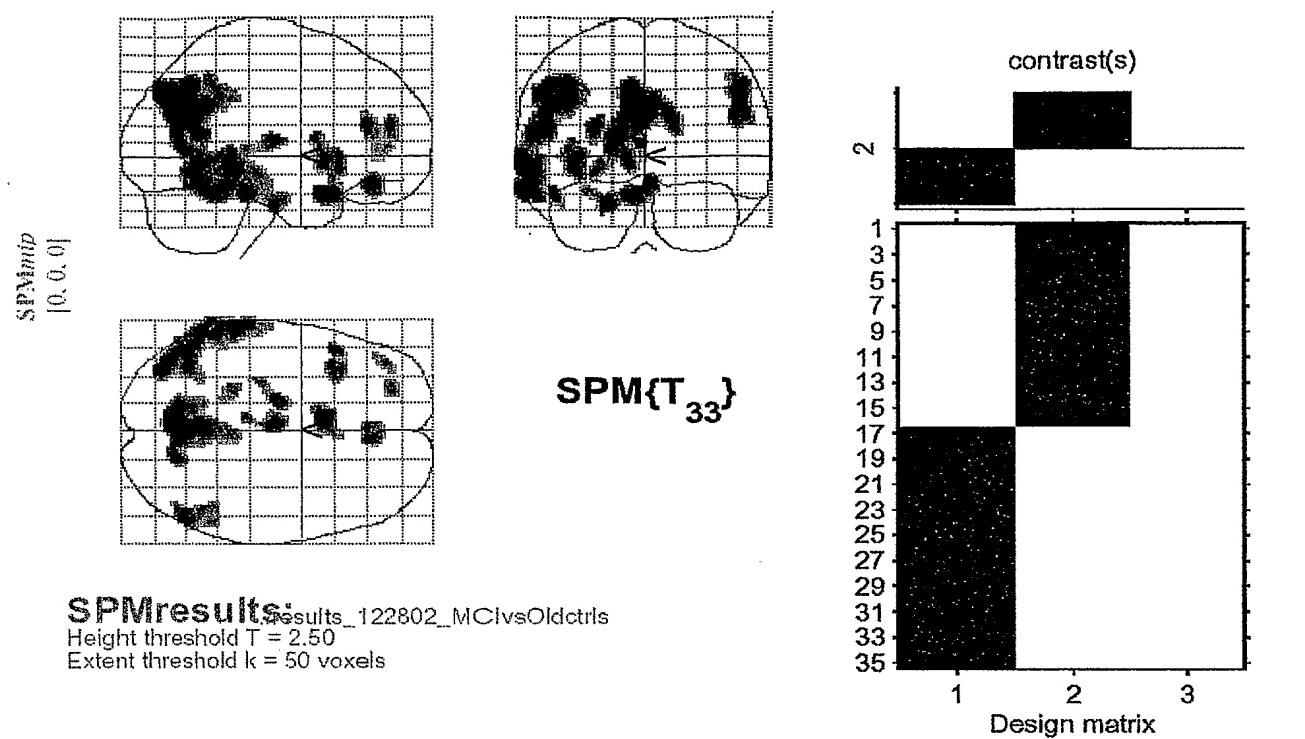
↓

720 The network is then trained on each input dataset
 ↓

730 The resulting trained CDInn is tested on the full dataset to assess its classification
accuracy.

FIG. 13FIG. 3AFIG. 14

**Fig. 3B**

**Statistics: volume summary (p-values corrected for entire volume)**

set-level	c	cluster-level			voxel-level			x,y,z {mm}
		p _{corrected}	k _F	p _{uncorrected}	p _{corrected}	T	(Z)	
0.173	14	0.000	1945	0.000	0.106	5.29	(4.47)	0.000 -4 -70 30
					0.483	4.54	(3.97)	0.000 -14 -68 16
					0.616	4.37	(3.86)	0.000 -4 -58 28
		0.000	2610	0.000	0.788	4.16	(3.70)	0.000 -42 -74 36
					0.804	4.13	(3.68)	0.000 -56 -56 16
					0.881	4.01	(3.59)	0.000 -60 -56 -8
		0.862	202	0.064	0.883	4.01	(3.59)	0.000 -6 -14 -24
	0.247	450	0.009		0.951	3.84	(3.47)	0.000 52 -64 38
					1.000	2.96	(2.77)	0.003 54 -50 44
		1.000	58	0.302	0.998	3.49	(3.20)	0.001 -48 18 -22
				0.991	114	0.153	(3.19)	0.001 -4 -12 10
					1.000	2.81	(2.64)	0.004 -8 -30 4
		0.996	101	0.177	0.999	3.43	(3.15)	0.001 4 -38 -16
	0.944	162	0.093		1.000	3.37	(3.10)	0.001 -10 14 -2
					1.000	2.90	(2.71)	0.003 -12 8 12
		0.981	131	0.127	1.000	3.32	(3.07)	0.001 -20 -14 -28
					1.000	2.91	(2.72)	0.003 -30 -22 -20
	0.888	191	0.071		1.000	3.22	(2.98)	0.001 -38 20 -2
	0.997	95	0.189		1.000	3.21	(2.97)	0.001 -26 48 16
	0.993	110	0.160		1.000	3.05	(2.84)	0.002 -2 36 18
					1.000	2.69	(2.54)	0.005 -2 42 2
	1.000	69	0.261		1.000	3.00	(2.80)	0.003 -24 -38 -6
				1.000	2.93	(2.74)	0.003 -40 46 6	

table shows at most local maxima > 8.0mm apart per cluster

Height threshold: T = 2.50, p = 0.009 (1.000 corrected)

Extent threshold: k = 50 voxels, p = 0.338 (1.000 corrected)

Expected voxels per cluster, $\langle k \rangle = 58.849$ Expected number of clusters, $\langle c \rangle = 10.48$

Degrees of freedom = [1,0,33,0]

Smoothness FWHM = [3.5, 13.7, 16.0] {mm} = 6.7 6.9 8.0 {voxels}

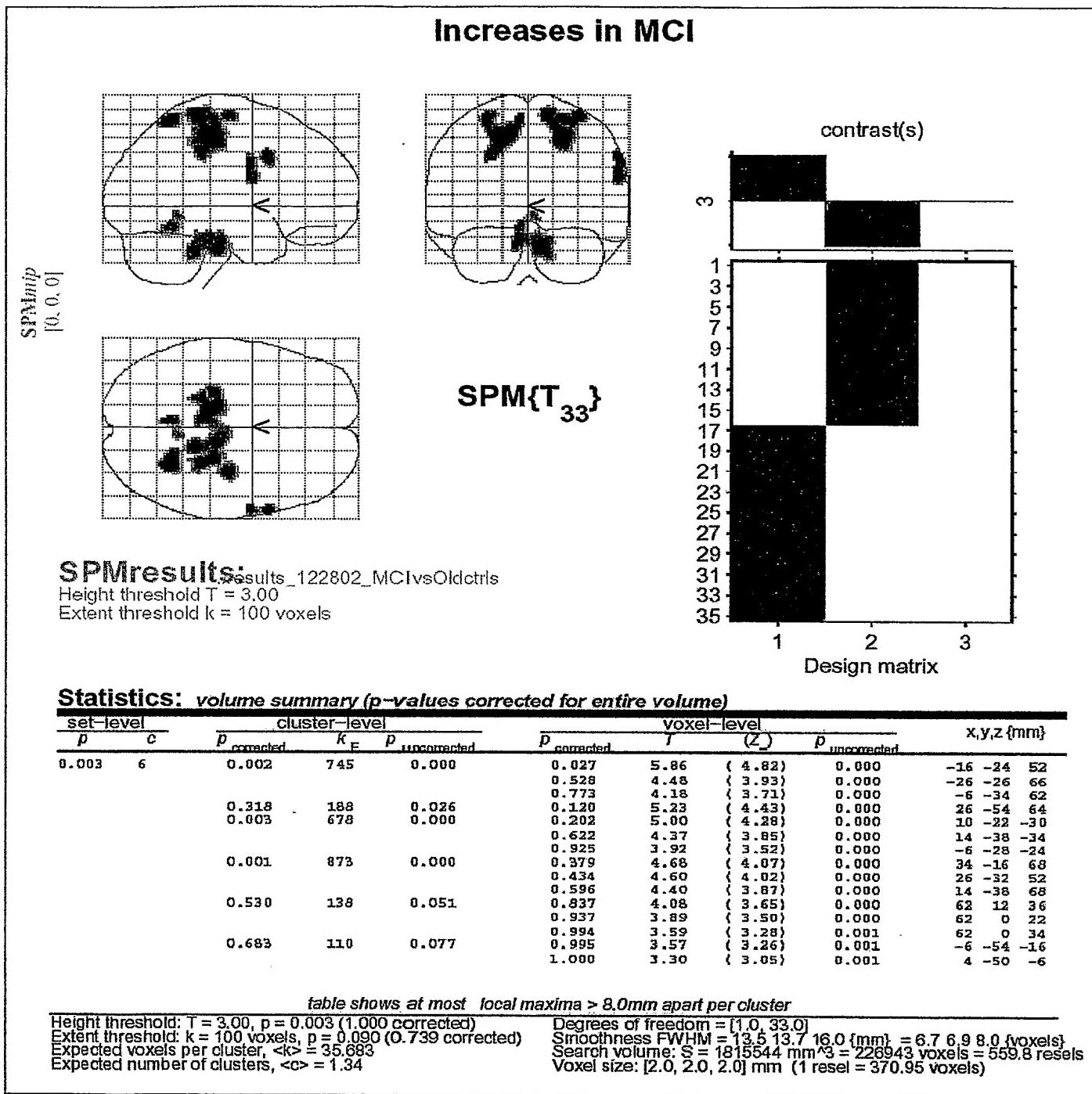
Search volume: S = 1815544 mm³ = 226943 voxels = 559.8 resels

Voxel size: [2.0, 2.0, 2.0] mm (1 resel = 370.95 voxels)

Statistics: single cluster summary (p-values corrected for entire volume)

cluster-level	k _F	p _{uncorrected}	voxel-level			x,y,z {mm}
			p _{corrected}	T	(Z)	
0.000	2610	0.000	0.788	4.16	(3.70)	0.000 -42 -74 36
			0.804	4.13	(3.68)	0.000 -56 -56 16
			0.881	4.01	(3.59)	0.000 -60 -56 -8
			0.892	3.99	(3.58)	0.000 -62 -36 -6
			0.905	3.96	(3.56)	0.000 -64 -30 -22
			0.938	3.89	(3.50)	0.000 -58 -46 -22
			0.977	3.74	(3.39)	0.000 -50 -60 42
			0.987	3.66	(3.33)	0.000 -42 -62 44
			0.998	3.50	(3.21)	0.001 -52 -62 28
			1.000	3.03	(2.83)	0.002 -56 -46 36
			1.000	3.01	(2.81)	0.002 -54 -68 16
			1.000	2.99	(2.79)	0.003 -62 -24 -8
			1.000	2.83	(2.66)	0.004 -62 -18 -24
			1.000	2.76	(2.60)	0.005 -32 -56 40

FIG. 4

**Fig. 5**

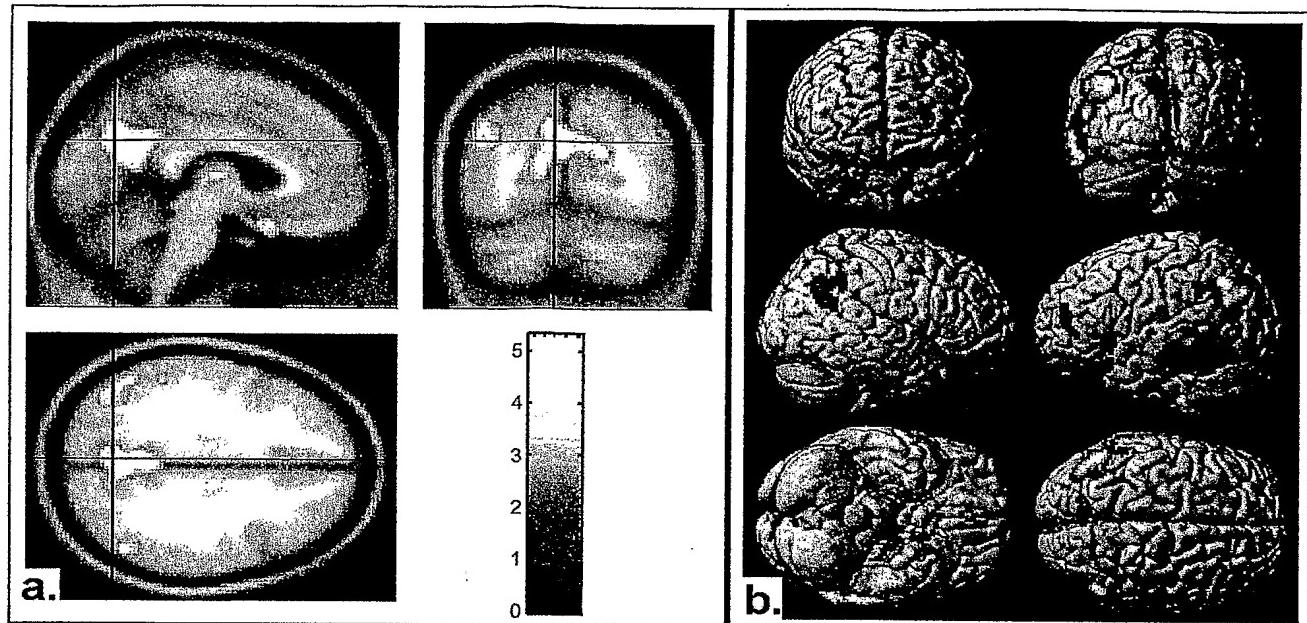


Fig. 6

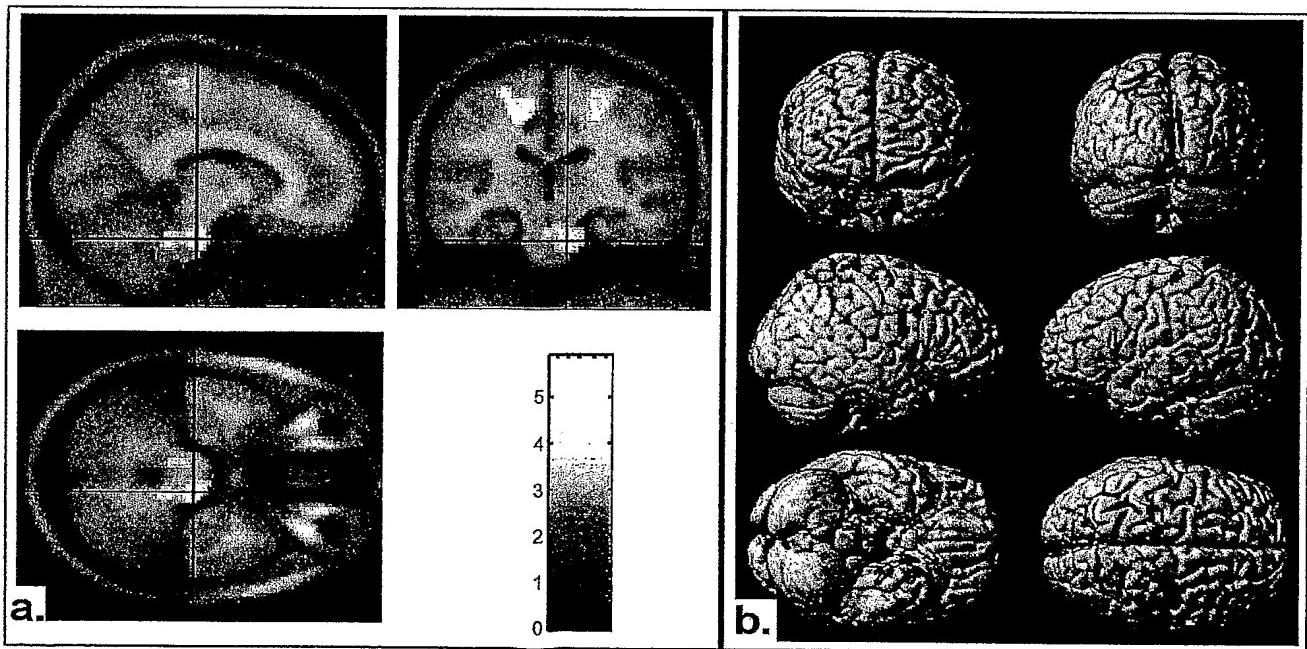


Fig. 7

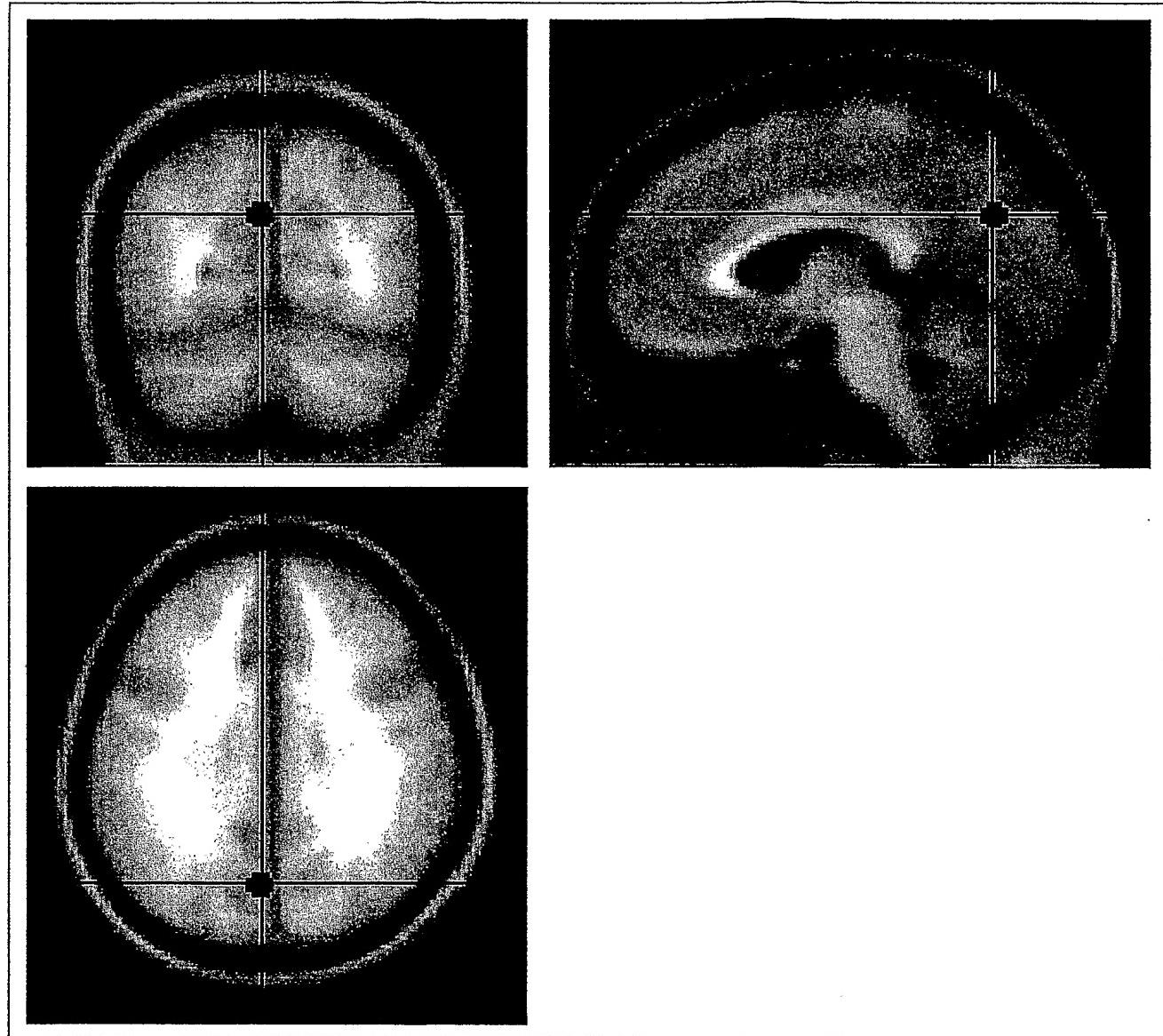


Fig. 8

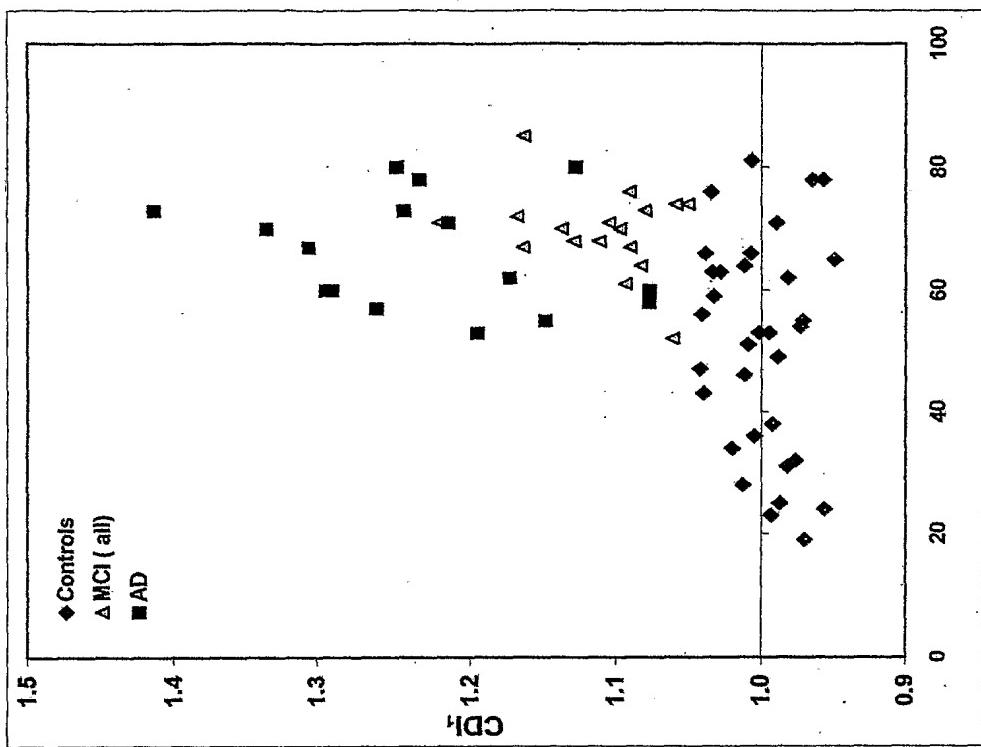


FIG. 10

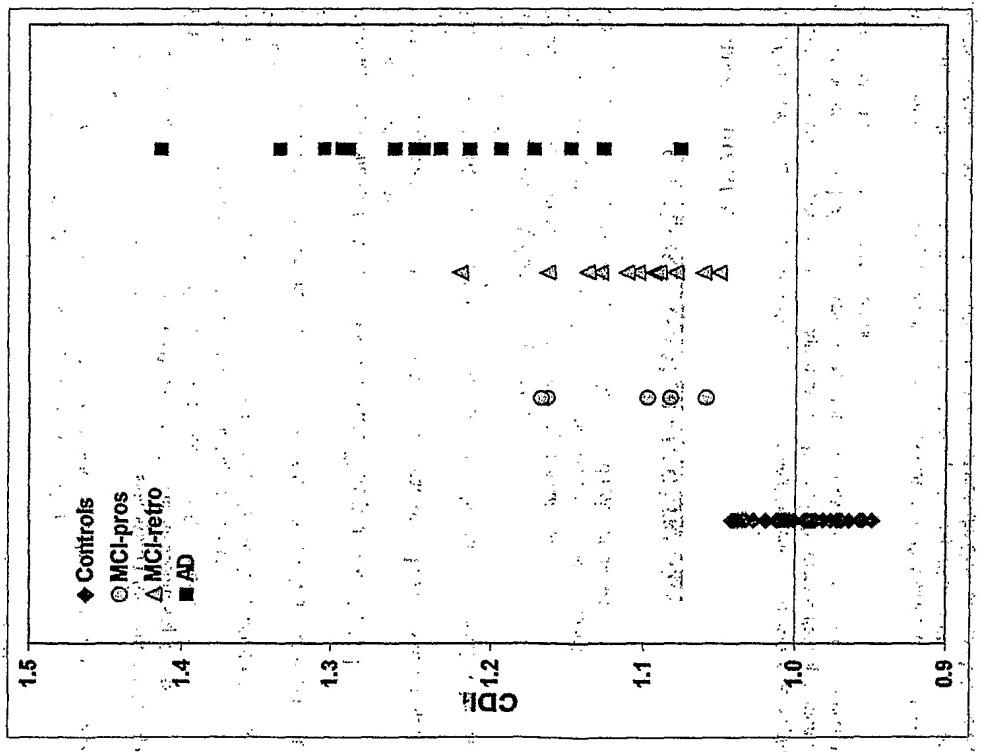
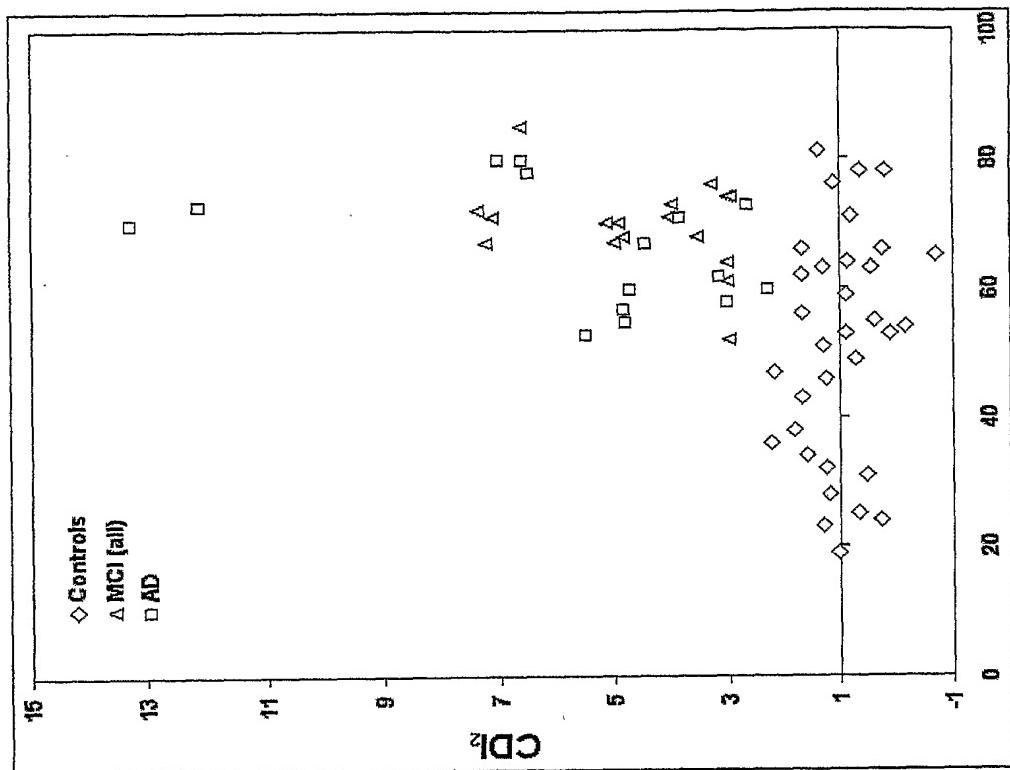
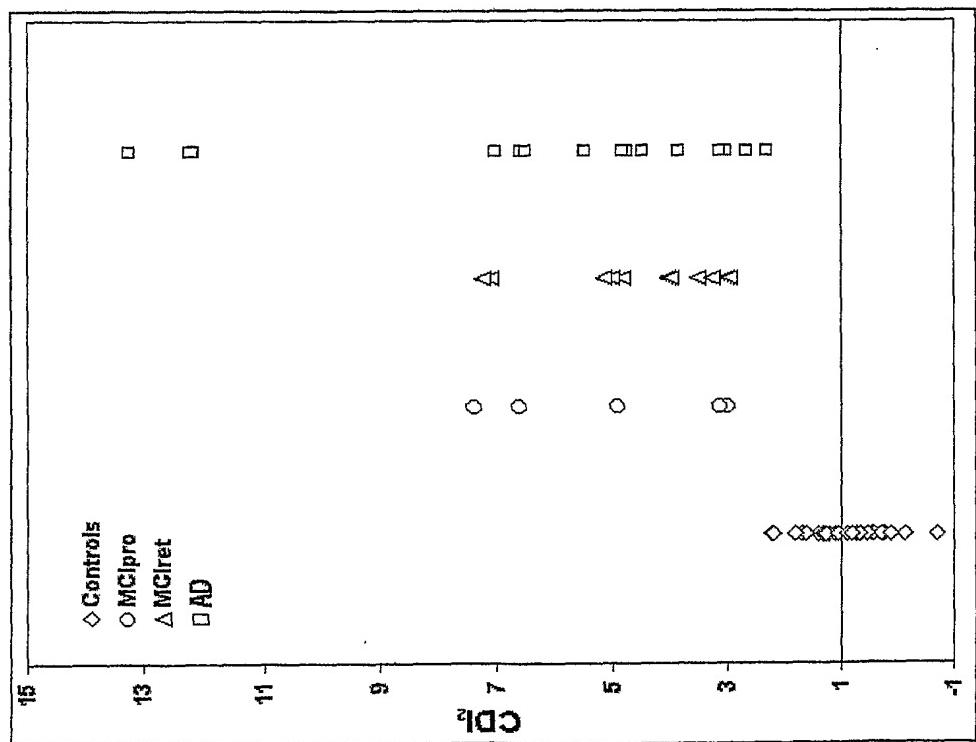


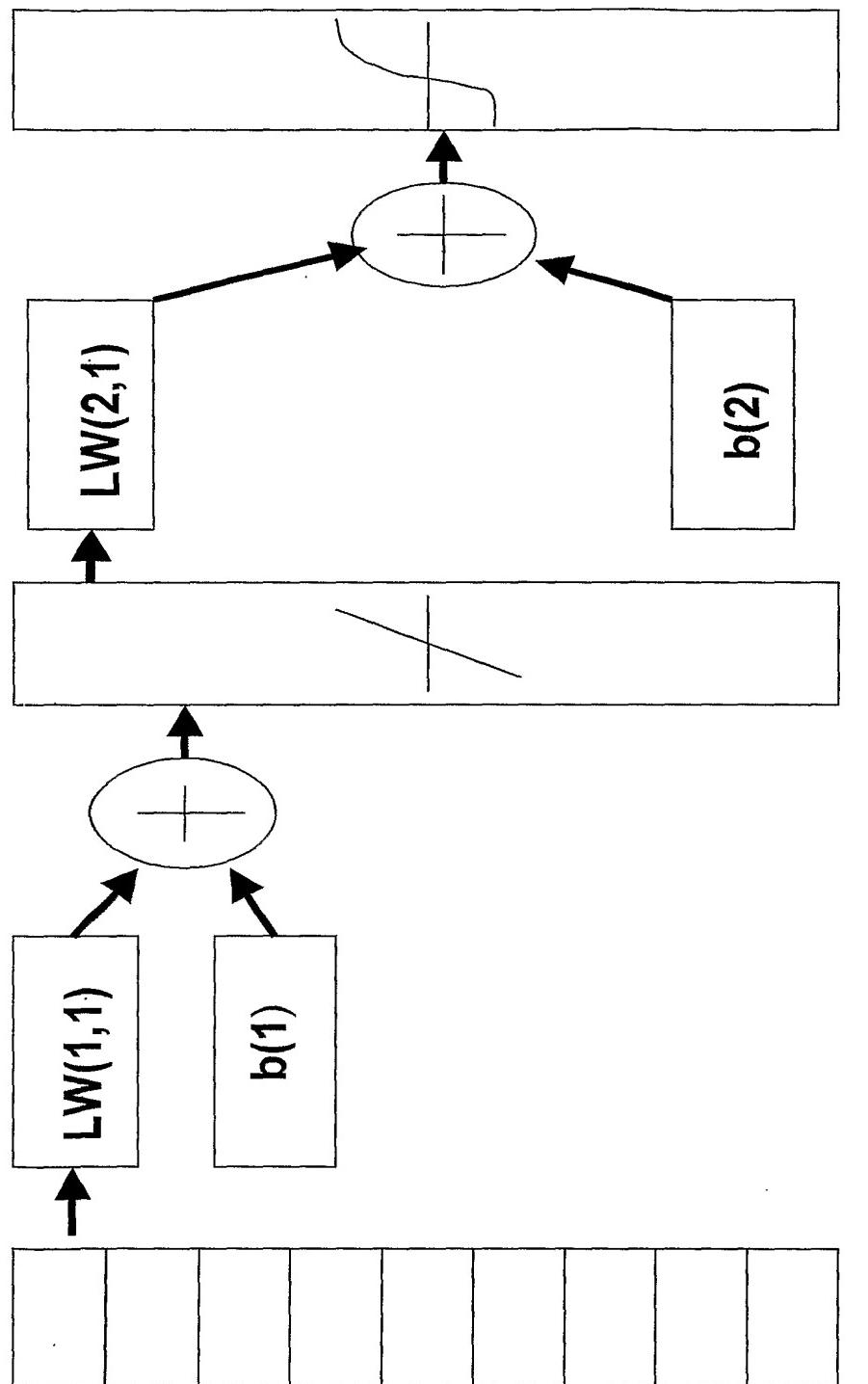
FIG. 9



F16.12



F16.11



9 Intensity Inputs

FIG. 15

1
1